



Un modèle de connaissances en RDF(S) pour raisonner à partir de cas sur le Web sémantique

Mathieu d'Aquin

► To cite this version:

Mathieu d'Aquin. Un modèle de connaissances en RDF(S) pour raisonner à partir de cas sur le Web sémantique. Atelier raisonnement à partir de cas - Plate-forme AFIA, Jean Lieber, Jul 2003, Laval, 9 p. inria-00099480

HAL Id: inria-00099480

<https://inria.hal.science/inria-00099480>

Submitted on 26 Sep 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Un modèle de connaissances en RDF(S) pour raisonner à partir de cas sur le Web sémantique

Mathieu d'Aquin

Orpailleur, LORIA, UMR 7503
CNRS, INRIA-Lorraine, Universités de Nancy
Campus scientifique - BP 239
54506 Vandœuvre-lès-Nancy
daquin@loria.fr

Résumé : Le Web sémantique vise à rendre les ressources du Web exploitables par des agents logiciels. Pour l'application KASIMIR, on souhaite pouvoir profiter de cette infrastructure dans le cadre du raisonnement à partir de cas (RÀPC). Nous développons dans cette optique un modèle des connaissances manipulées par le processus de RÀPC qui soit adaptable aux ressources diffusées sur le Web sémantique. Ce modèle se place dans le cadre des reformulations et s'appuie sur le langage RDF(S), standard pour la représentation des connaissances sur le Web sémantique. Il sera utilisé pour développer un service de RÀPC exploitant notamment les ressources liées à l'application KASIMIR.

Mots-clés : reformulation, chemin de similarité, Web sémantique, RDF(S).

1 Introduction

Le Web sémantique (Berners-Lee *et al.*, 2001) se veut une extension du Web actuel dans laquelle les ressources, informations et connaissances sont décrites de façon formelle, pour être exploitables par des agents logiciels. Ainsi, un grand nombre de travaux se concentrent sur le développement de nouvelles technologies issues des domaines de l'intelligence artificielle dans le contexte du Web. Ces technologies doivent répondre aux problèmes caractéristiques de ce contexte — le passage à l'échelle, l'hétérogénéité et la distributivité des sources d'informations et de connaissances, etc. — par un souci constant de standardisation, de modularité et d'interopérabilité.

Le projet KASIMIR, qui est ici notre première motivation, a pour objectif la gestion des connaissances en cancérologie (Lieber *et al.*, 2002). Les bases de connaissances sur lesquelles s'appuie le système sont appelées *référentiels* et sont vues comme la représentation de guides de bonnes pratiques en cancérologie. Les référentiels décrivent les traitements à appliquer dans les cas les plus courants, notamment pour le cancer du sein. Le but de cette gestion des connaissances est l'homogénéisation et l'amélioration de la qualité des soins par la diffusion et l'accès aux connaissances des référentiels pour des médecins en consultation. Mais les référentiels ne permettent pas de répondre à tous les cas. Les solutions proposées doivent souvent être *adaptées* pour des cas particuliers (les patients ayant des contre-indications aux traitements proposés, par exemple). Ainsi, une autre étape du développement de KASIMIR est l'utilisation du raisonnement à partir de cas (RÀPC) pour adapter les solutions proposées par les référentiels aux cas qui sortent du cadre des référentiels.

L'infrastructure du Web sémantique semble bien adaptée pour répondre à la problématique de diffusion des connaissances de KASIMIR (Brachais *et al.*, 2003). Malheureusement, très peu de travaux portent sur l'application du RÀPC dans ce cadre (Champin, 2002). Le but de ce papier est d'étudier les principes pour l'implantation d'un raisonneur à partir de cas utilisable sur les ressources du Web sémantique. Cela passe par la réalisation d'un modèle générique des connaissances manipulées par le RÀPC, adapté aux technologies standards du Web sémantique. Les fondements du modèle proposé sont issus des travaux sur les *reformulations* (Lieber, 1999; Melis *et al.*, 1998), représentées dans le cadre du langage RDF(S), un des langages de prédilection pour la représentation des connaissances sur le Web sémantique. Notre objectif est d'exploiter ce modèle pour construire un service capable de raisonner à partir de cas sur les connaissances diffusées au travers du futur portail sémantique KASIMIR (Brachais *et al.*, 2003). Nous ne prétendons donc pas présenter ici un développement

abouti, notre ambition étant plus de discuter les principes, les difficultés et les intérêts d'un tel développement, ainsi que les interrogations qu'il suscite.

Pour cela, quelques unes des technologies développées dans le cadre du Web sémantique sont présentées section 2. Les reformulations sont introduites à la section 3 et la section 4 revient sur les raisons du choix des reformulations comme modèle pour le RÀPC et de RDF(S) comme langage de représentation pour ce modèle. La section 5 détaille la mise en œuvre des reformulations en RDF(S) et la section 6 décrit quelques unes des difficultés et des approches envisagées pour construire un raisonneur à partir de cas manipulant le modèle des reformulations sur les ressources du Web sémantique.

2 Le Web sémantique

Le Web sémantique consiste à étendre le Web actuel, principalement composé de textes en langue naturelle et d'images, vers un Web contenant des informations avec lesquelles des agents logiciels puissent raisonner. Il s'agit d'établir les technologies et les usages pour un « Web de connaissances ». Cette section présente quelques unes de ces technologies, en particulier celles utiles au développement d'un service de raisonnement : les langages de représentation¹ et les raisonneurs sur ces langages.

Langages pour le Web sémantique.

Le développement du Web sémantique nécessite de disposer de langages ayant une syntaxe et une sémantique standards et formellement décrites (Laublet *et al.*, 2002). Le langage XML (*eXtensible Markup Language*) est à l'heure actuelle le standard incontournable pour l'échange d'informations sur le Web. Les DTD (*Document Type Definition*) et les schémas XML permettent de contraindre l'utilisation des balises et de leurs imbrications pour représenter les informations. De cette façon, XML permet de déclarer et de partager une syntaxe spécifique à un domaine ou une application. Mais, ne s'intéressant qu'à la syntaxe et non à la sémantique, XML ne peut être considéré comme un langage de représentation des connaissances (Decker *et al.*, 2000).

RDF (*Resource Description Framework*) est décrit comme un langage pour la définition de métadonnées sur les documents du Web. Il permet de déclarer des triplets (objet, prédicat, sujet) où les objets et les sujets sont des ressources du Web et les prédicats des relations entre ces ressources. RDF possède une syntaxe XML accédant ainsi à ses aspects de standard adapté à la diffusion sur le Web. RDF SCHÉMA est le langage de définition de vocabulaires pour RDF. Il permet de définir des hiérarchies de classes et de propriétés (relations entre classes), ainsi que certaines contraintes simples sur les propriétés. Les ressources de RDF sont les instances des classes de RDF SCHÉMA et doivent respecter ces contraintes. De plus, RDF SCHÉMA, en tant que langage de représentation des connaissances, possède certaines fonctionnalités intéressantes. D'une part, il est réflexif : les objets de son méta-modèle, l'objet qui représente les classes par exemple, peuvent être manipulés au sein du langage. D'autre part, il permet la réification de propriétés. En effet, les propriétés sont considérées comme des classes et peuvent ainsi posséder elle même des propriétés. Enfin, RDF SCHÉMA hérite de RDF sa syntaxe XML. Tout cela contribue à faire de RDF(S)² un premier formalisme de représentation des connaissances sur le Web sémantique. Il possède d'ailleurs une sémantique standard³ définissant les inférences possibles sur le langage.

Par sa simplicité et la souplesse de son modèle, RDF(S) est conçu pour être étendu et servir de base à des langages plus expressifs. OWL (*Web Ontology Language*) est une de ces extensions. Il implémente une logique de descriptions très expressive (de type *SHIQ*) dans un schéma RDF(S). Un principe pour une telle extension est que les constructions de RDF(S) et les inférences sur ces constructions doivent rester valables dans le nouveau langage. C'est une extension autant sémantique que syntaxique. Notons que OWL fait suite au langage DAML+OIL, qui lui même est issu du langage OIL.

¹Notamment ceux développés dans le cadre du W3C (XML, RDF, RDF SCHÉMA et OWL) dont les spécifications sont accessibles à partir de www.w3.org.

²Par la suite on notera RDF(S) les technologies RDF et RDF SCHÉMA utilisées conjointement.

³www.w3.org/TR/rdf-mt/

Raisonnements sur ces langages.

Définir des langages expressifs ayant une sémantique formelle et partagée est une chose, encore faut-il disposer des outils capables de mettre en œuvre cette sémantique (Lassila, 2002). Vus les développements sur les langages de représentation d'ontologies, les logiques de descriptions semble devoir jouer un rôle prédominant dans ce domaine. Les systèmes tels que FACT (Horrocks, 1999) et RACER (Haarslev & Möller, 2001), qui raisonnent sur des logiques de descriptions de type \mathcal{SHIQ} , sont d'ores et déjà utilisés pour des applications dédiées aux langages du Web sémantique. C'est le cas par exemple dans l'éditeur d'ontologies OILED⁴ où le raisonnement vient appuyer l'aide à l'édition de connaissances.

D'autres raisonneurs plus simples sont développés dans le but de réaliser les inférences prévues par la sémantique de RDF(S). Ces systèmes doivent respecter les principes de modularité et d'extensibilité qu'impose le cadre du Web sémantique. Le système TRIPLE (Sintek & Decker, 2002), par exemple, permet de réaliser des inférences sur des données et des schémas décrits en RDF(S), sans pour autant utiliser une sémantique fixe et prédéfinie pour ce langage. Celle-ci est déclarée à l'aide de règles d'inférences logiques qui seront vérifiées par un raisonneur externe. Il est ainsi possible de choisir quelles parties de la sémantique de RDF(S) on souhaite utiliser et surtout, il est possible d'étendre cette sémantique pour des schémas spécifiques à un domaine ou à une application.

Il convient de remarquer que les langages et les raisonneurs cités sont conçus pour des raisonnements de type déduction logique et classification. Ces modes d'inférences nécessitent un modèle de connaissances du domaine complet et bien défini (sous la forme d'une ontologie en OWL par exemple). Cela peut sembler difficile dans le contexte du Web, naturellement complexe et peu structuré. Le RÀPC, en plus d'être utile à l'application KASIMIR, semble pouvoir jouer un rôle dans ce cadre. Malheureusement, très peu de travaux s'intéressent à l'application du RÀPC sur le Web sémantique (Champin, 2002) et nous n'en connaissons aucun qui, comme nous, ont pour but de construire un raisonneur à partir de cas générique, applicable aux ressources du Web sémantique.

3 Les reformulations

Le RÀPC s'attache à résoudre un problème cible, noté *cible*, en utilisant la solution $Sol(srce)$ d'un problème source *srce* déjà résolu et similaire à *cible*. Le résultat attendu est la solution $Sol(cible)$ du problème cible. L'étape de recherche dans une base de cas du problème source est appelée *remémoration* et l'étape qui transforme la solution du problème remémoré en une solution qui convienne au problème à résoudre est appelée *adaptation* (voir figure 1).

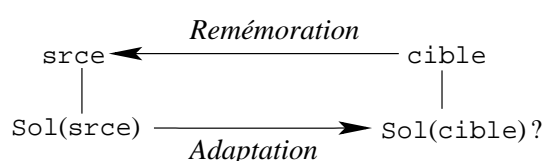


FIG. 1 – La remémoration et l'adaptation en RÀPC.

Afin d'effectuer ces étapes, un système de RÀPC doit disposer d'un modèle lui permettant de représenter les objets qu'il manipule. En particulier, il doit être en mesure de décrire les problèmes et les solutions, de disposer d'une représentation de la similarité entre problèmes et des éléments d'adaptation des solutions. De plus, la similarité entre problèmes doit servir de guide à l'adaptation des solutions et dans le cadre de la *remémoration guidée par l'adaptation* (*adaptation-guided retrieval* (Smyth, 1996)), les connaissances d'adaptation doivent servir de guide à la remémoration. Les liens existant entre similarité de problèmes et adaptation des solutions doivent donc eux aussi être représentés dans un modèle générique, adaptable à des domaines d'application spécifiques. C'est dans ce cadre qu'ont été définies les *reformulations* (Lieber, 1999; Melis *et al.*, 1998).

⁴À l'origine conçu pour le langage OIL, puis étendu à DAML+OIL et bientôt à OWL (Bechhofer *et al.*, 2001).

Une reformulation est un couple (r, \mathcal{A}_r) , où r est une relation entre problèmes et \mathcal{A}_r est une adaptation de solutions. Une *relation entre problèmes* est un élément symbolique pour représenter la similarité entre deux problèmes. Deux problèmes `cible` et `srce` sont dit similaires relativement à r (noté $srce \ r \ cible$) si leur unique différence est capturée par la relation r . La relation la plus évidente est certainement l'équivalence (\equiv) qui dénote le lien existant entre deux problèmes décrits de la même façon. Un *chemin de similarité* représente une similarité complexe (composée) entre problèmes. Un chemin de similarité entre `srce` et `cible` est une séquence de relations r_i allant de `srce` à `cible` via des problèmes intermédiaires pb_i :

$$cible = pb_0 \ r_1 \ pb_1 \ r_2 \ pb_2 \ \dots \ pb_{q-1} \ r_q \ pb_q = srce$$

Les *adaptations de solutions* sont relatives aux relations entre problèmes et sont utilisées pour transformer une solution en une autre. Comme pour les chemins de similarité, un *chemin de modification* représente une transformation complexe de solutions par une séquence d'adaptations de solutions. Finalement, une *reformulation* (r, \mathcal{A}_r) peut être vue comme une règle d'adaptation permettant de déclarer que si deux problèmes sont en relation par r , la solution du premier peut être adaptée par \mathcal{A}_r pour obtenir la solution du second. La reformulation la plus évidente est certainement $(\equiv, copie)$ signifiant que si deux problèmes sont équivalents, la copie de la solution du premier résout le second.

Ainsi, pour raisonner à partir de cas en utilisant le modèle des reformulations, l'étape de remémoration consiste à trouver un chemin de similarité entre le problème `cible` à résoudre et un problème `srce` de la base de cas. L'adaptation consiste en la construction du chemin de modification en suivant en sens inverse le chemin de similarité et en appliquant les reformulations correspondantes. Sur l'exemple de la figure 2, le problème `cible` correspond à une patiente atteinte d'un cancer du sein. Le référentiel correspondant ne permet pas de répondre à ce cas pour deux raisons : elle est diabétique, ce qui entraîne une contre-indication au tamoxifène (produit d'hormonothérapie) et on ne sait pas si des ganglions ont été envahis par des cellules cancéreuses. Ainsi, l'étape de remémoration construit un chemin de similarité composé des deux relations entre problèmes (r_1 et r_2) pour trouver le patient `srce` similaire à `cible` et dont on connaît le traitement⁵. La solution $Sol(srce)$ ainsi obtenue doit ensuite être adaptée pour construire la solution du problème posé. Pour cela le chemin de modification est créé en appliquant les deux reformulations correspondant aux relations du chemin de similarité. On obtient de cette façon la solution $Sol(cible)$ correspondant au traitement $Sol(srce)$ duquel on a enlevé le tamoxifène en raison de la contre-indication.

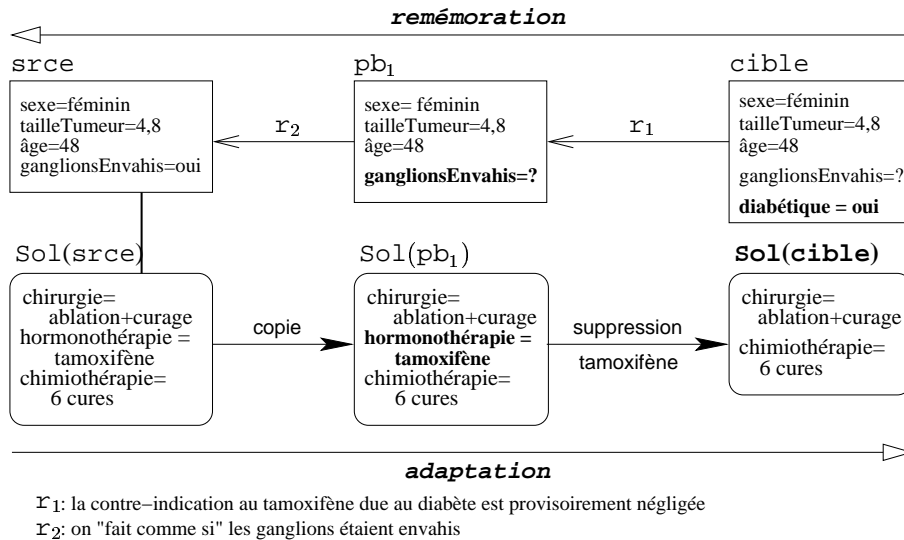


FIG. 2 – Exemple d'utilisation de deux reformulations pour le RÀPC.

⁵On remarque que lorsque l'on ignore si les ganglions sont envahis ou non, on « fait comme si » ils l'étaient envahis car cela entraîne un geste chirurgical plus large (curage en plus de l'ablation) (Lieber *et al.*, 2001).

4 Choix du modèle et du langage

Dans cette section nous revenons sur les raisons du choix du modèle des reformulations et de son implantation en RDF(S) pour représenter les objets manipulés par le RÀPC sous la forme de ressources du Web sémantique.

4.1 Le choix des reformulations

Vouloir implanter un raisonneur à partir de cas sur les ressources du Web sémantique suppose de disposer d'un modèle générique pour décrire les objets et les notions du RÀPC. Outre le fait que nous le connaissons bien, nous avons choisi le modèle des reformulations pour plusieurs raisons. Ce modèle représente les notions générales manipulées par le RÀPC, comme la similarité et l'adaptation, sous une forme exclusivement symbolique. En effet, les reformulations ont été à l'origine développées pour comparer des structures complexes (formules logiques chez (Melis, 1995) et graphes moléculaires chez (Lieber, 1997)) et étudiées en particulier dans le cadre de la représentation des connaissances par objets. De plus, le modèle des reformulations est fondé sur la réification de relations complexes entre objets : les relations entre problèmes, les adaptations de solutions et finalement les reformulations. Les langages du Web, en particulier RDF(S), accordent un rôle central aux liens existant entre les ressources. Enfin, parce qu'il décrit de façon simple et très générale les notions et objets essentiels du RÀPC, le modèle des reformulations peut effectivement être considéré comme générique. Il est ainsi utilisable pour des domaines d'application variés et est extensible, en particulier, pour inclure d'autres modèles et d'autres approches pour le RÀPC.

Le principal désavantage à utiliser le modèle des reformulations est que, en se plaçant dans un cadre aussi général et générique, il est nécessaire de mettre en œuvre un grand nombre de connaissances pour spécialiser le modèle dans une application particulière. La construction d'un système de RÀPC utilisant les reformulations entraîne ainsi un effort important d'acquisition et de représentation des connaissances. L'utilisation des principes et des technologies du Web sémantique devrait nous faciliter la tâche en nous donnant accès à la somme importante de connaissances disponibles sur le Web.

4.2 Le choix de RDF(S)

Pouvoir appliquer le modèle des reformulations aux ressources du Web sémantique suppose que les deux parties (modèle et ressources) soient décrites dans des langages ayant une base commune. En tant que standard pour la représentation sur le Web sémantique, RDF(S) s'impose naturellement. En effet, par sa simplicité et la souplesse de son modèle, RDF(S) est en passe de devenir, en pratique, le langage de référence pour le développement de bases de connaissances et d'informations sur le Web sémantique. De plus, il est conçu et utilisé comme fondement pour d'autres langages de représentation d'ontologies, plus expressifs et plus complexes. De fait, développer un modèle à partir de RDF(S) permet d'en garantir la compatibilité avec toutes ses extensions. Le modèle des reformulations permettra ainsi de raisonner sur des ressources non seulement en RDF(S), mais aussi en OWL, en DAML+OIL, etc. Nous pourrions, en outre, profiter de l'extensibilité de RDF(S) pour améliorer le modèle avec d'autres approches du RÀPC ou d'autres mécanismes de représentation. Par ailleurs, le modèle simple et proche de la représentation des connaissances par objets de RDF(S) le rend particulièrement bien adapté pour représenter les reformulations. Les fonctionnalités qu'il met à disposition pour la représentation des connaissances, notamment la réification de relations, s'avère très utile dans ce cadre. Enfin, l'aspect réflexif de RDF(S) permet de considérer les objets décrits dans le modèle eux même comme des ressources. Les similarités et les adaptations issues du raisonnement deviendront elles-mêmes sujettes à traitement et à raisonnement sur le Web sémantique.

5 Les reformulations en RDF(S)

La figure 3 présente le modèle de reformulations tel qu'implanté en RDF(S) selon quatre « niveaux » : la résolution de problèmes, la similarité, l'adaptation et les reformulations. Ces quatre niveaux sont détaillés dans cette section, illustrés de leur utilisation sur l'exemple de KASIMIR.

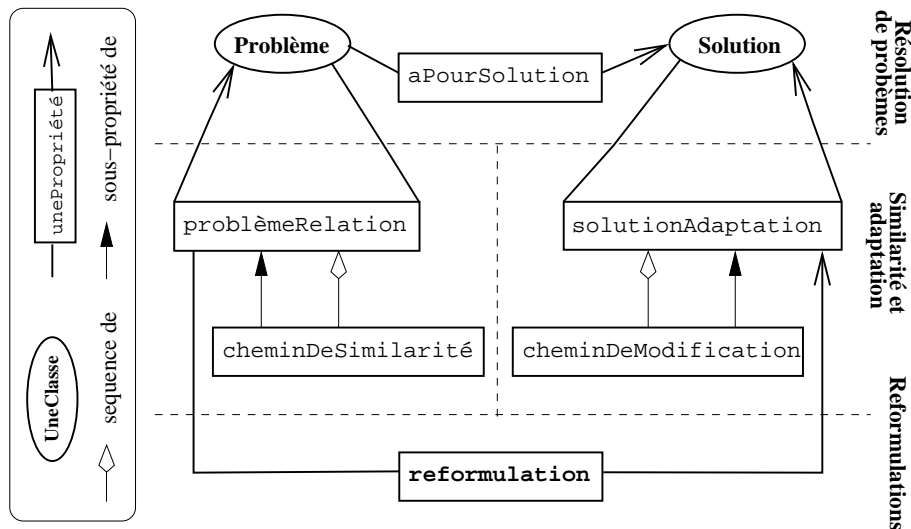


FIG. 3 – Le modèle des reformulations tel que représenté en RDF(S).

Les classes, les propriétés et la relation de sous-propriété sont des objets directement utilisables du langage RDF(S) (`rdfs:Class`, `rdf:Property` et `rdfs:subPropertyOf`). Concernant les séquences, celles-ci sont implantées par l'intermédiaire de la classe `rdf:Seq` de RDF(S). Les propriétés `cheminDeSimilarité` et `cheminDeModification` sont des sous-classes de `rdf:Seq`. Rappelons que cela est rendu possible par les caractéristiques particulières du langage RDF(S) (réflexivité et réification).

Résolution de problèmes.

Ce niveau contient les objets génériques permettant de décrire les problèmes, les solutions et la relation `aPourSolution` qui associe un problème à sa solution. Ainsi, la base de cas pourra être représentée par un ensemble de triplets de type (`Problème`, `aPourSolution`, `Solution`). Par exemple dans KASIMIR, les problèmes correspondent à des descriptions de patients atteints de cancer et les solutions sont les traitements à appliquer pour les soigner. Une description en RDF des référentiels associe des objets de la classe `Patient` à des objets de la classe `Traitement` par une propriété `aPourTraitement`. Il suffira donc de déclarer les liens d'instanciation (`rdfs:type`) adéquats entre la partie résolution de problèmes du modèle et les objets des référentiels pour qu'un raisonneur utilisant les reformulations considère un référentiel comme une base de cas.

Similarité et adaptation.

Ces deux parties du modèle contiennent les objets génériques permettant de décrire les éléments de la similarité et de l'adaptation. Les éléments simples pour représenter ces notions dans le cadre des reformulations correspondent aux relations entre problèmes et aux adaptations de solutions. Ils sont représentés par des propriétés générales, respectivement entre problèmes et entre solutions. Celles-ci pourront ensuite être spécialisées en sous-propropriétés dans le domaine d'application (`contreIndicationAuTamoxifene`, `suppressionTamoxifene`, etc.).

Une similarité complexe correspond à une composition de relations entre problèmes. Elle est représentée par la propriété `cheminDeSimilarité` qui est une séquence de relations entre problèmes. On remarque que `cheminDeSimilarité` est définie comme une sous-propropriété de `problèmeRelation` signifiant ainsi qu'une composition de relations entre problèmes est aussi considérée comme une relation. Il en va de même pour les chemins de modification et les adaptations de solutions.

Reformulations.

Une reformulation peut être vue comme une règle d'adaptation qui associe une relation sur les problèmes à une adaptation de solutions. Les reformulations sont ainsi implantées sous la forme d'une propriété liant `problèmeRelation` à `solutionAdaptation`. On remarque que, parce que les chemins de similarité sont considérés comme des relations entre problèmes et les chemins de modification comme des adaptations de solutions, il sera possible de construire des connaissances d'adaptation complexes, portant sur des similarités ou des adaptations composées, par le biais des reformulations.

Le modèle défini permet de lier les objets du domaine d'application, décrits dans un formalisme issu de RDF, aux objets et notions du RÀPC. Un raisonneur mettant en œuvre ce modèle sera ainsi en mesure d'exploiter les ressources mises à disposition sur le Web sémantique⁶ pour raisonner à partir de cas. La section suivante décrit les approches envisagées et celle que nous pensons retenir, pour l'implantation d'un tel raisonneur *paramétrable par les ressources du Web sémantique*.

6 Approches envisagées pour l'implantation d'un service de RÀPC sur le modèle des reformulations

Pour implanter les procédures permettant de raisonner à partir de cas sur le modèle proposé, il semble naturel de vouloir conserver l'approche déclarative, modulaire et extensible préconisée par le Web sémantique. On peut par ailleurs considérer le modèle des reformulations comme une extension de RDF(S) pour le RÀPC. À ce titre, sa sémantique doit être définie et doit venir s'ajouter à celle de RDF(S). L'implantation d'un service RÀPC sur ce modèle passe par la mise en œuvre de sa sémantique au travers d'un raisonneur adapté. Il semble donc tout aussi naturel d'envisager la réutilisation et l'extension des technologies existantes liées aux raisonnements sur RDF(S). Mais, les procédures à mettre en œuvre pour manipuler les reformulations (la génération de chemins de similarité entre problèmes par exemple) sont complexes et difficiles à formaliser. Seules certaines inférences simples peuvent être décrites par l'intermédiaire des règles logiques qu'utilisent des outils comme TRIPLE. Par ailleurs, ces technologies (raisonneurs, éditeurs, etc.) sont encore à l'heure actuelle en cours de développement. Certaines fonctionnalités de représentation de RDF(S), comme la réification de propriétés, ne sont souvent pas prises en compte. De plus, la plupart de ces outils n'offrent pas les facilités nécessaires à leur réutilisation (API accessible, documentation, entrées/sorties exploitables, etc).

L'approche envisagée pour contourner ces difficultés consiste à construire un ensemble de services Web, chacun réalisant les inférences relatives aux différents niveaux du modèle. En effet, la mise en œuvre de technologies standards, s'appuyant sur XML, pour formaliser les échanges d'informations et la description de tels services est un point important de la construction du Web sémantique que nous n'avons pas encore abordé. Cette formalisation a pour objectif un accès transparent aux services Web en permettant l'automatisation de leur découverte, invocation et composition. L'utilisation des technologies relatives à cette problématique⁷ pour l'implantation d'un système de RÀPC sur le Web sémantique permettrait d'obtenir, en plus d'un modèle des connaissances, un modèle formel et opérationnel des tâches du RÀPC (Fuchs, 1997), toujours dans le cadre souple, extensible et modulaire que propose le Web sémantique. Il conviendra alors d'implanter un service pour chacune des tâches génériques du RÀPC, chaque service réalisant les inférences relatives aux objets d'un niveau particulier du modèle des reformulations (le service de remémoration, par exemple, manipule et génère les relations entre problèmes et les chemins de similarité). De cette façon, il sera possible d'ajouter et d'intégrer facilement des services pour prendre en compte d'autres approches du RÀPC ou des tâches spécifiques au domaine d'application. On conserve ainsi l'aspect extensible du modèle des connaissances pour les procédures qui le manipulent. De plus, certains des services

⁶Ces ressources pourront par ailleurs provenir de sources différentes. Par exemple, Kasimir pourra utiliser en plus de ses référentiels, les informations concernant les contre-indications aux traitements disponibles sur les sites des laboratoires pharmaceutiques correspondant.

⁷Citons le protocole pour l'échange d'informations SOAP, le langage de description de services WSDL et les efforts de normalisation des architectures centrées sur les services au W3C, ainsi que DAML-S, d'écrit comme une ontologie en DAML+OIL pour raisonner sur les services (www.daml.org/services/)

utiles peuvent d'ores et déjà s'appuyer sur les systèmes existants. Les inférences relatives à RDF(S) ou OWL, par exemple, pourront être réalisées par les raisonneurs cités en section 2.

7 Discussion

Nous avons présenté une approche pour construire un raisonneur à partir de cas sur les ressources du Web sémantique. Les avantages pour un système de RÀPC de s'intégrer au Web sémantique sont nombreux. En particulier, l'application KASIMIR pourra par ce biais profiter à la fois d'un mode de raisonnement souple et adaptable et de l'infrastructure, des technologies et surtout de la somme croissante de ressources qu'offre le Web sémantique. De plus, l'approche générique, modulaire et extensible imposée par le Web sémantique permet d'inclure facilement les connaissances spécifiques au domaine d'application, ainsi que d'autres approches du RÀPC. On pourra par exemple, construire, en lien avec le modèle des reformulations, un modèle pour représenter les *dépendances* entre problèmes et solutions (Fuchs *et al.*, 2001), ainsi que le service permettant de les mettre en œuvre. Par ailleurs, disposer d'un raisonneur à partir de cas souple et extensible sur les ressources du Web sémantique nous permettra aussi d'utiliser le RÀPC pour répondre à certaines problématiques du Web sémantique. En particulier, dans notre approche, le RÀPC manipule des connaissances provenant de diverses sources d'informations, de diverses ontologies. Le modèle des reformulations permet d'attribuer à chacune des ressources disponibles au travers de ces ontologies, un rôle dans le raisonnement (problèmes, solutions, similarité, adaptation, etc.). Il est ainsi possible de répondre à des requêtes complexes, de raisonner, de créer de nouvelles connaissances, par la collaboration de connaissances extérieures, pas nécessairement prévues pour « marcher ensemble ». Ceci rejoint une des problématiques parmi les plus importantes et les plus difficiles du Web sémantique : *l'interopérabilité des informations et des ontologies*. Le modèle et l'approche proposés, en mettant le RÀPC au centre de l'interopérabilité, pourront apporter une part de réponse à cette problématique.

8 Conclusion et perspectives

Nous avons présenté un modèle des objets manipulés par le processus de RÀPC fondé sur les reformulations. Les reformulations permettent de disposer d'une représentation symbolique et paramétrable par les connaissances du domaine, des notions de similarité et d'adaptation, ainsi que des connaissances pour les mettre en relation. Nous avons mis en œuvre ce modèle dans le langage RDF(S). C'est un langage souple, extensible et possédant des fonctionnalités utiles à la représentation de ce type de modèle. Il est considéré comme standard pour la représentation des connaissances sur le Web, permettant ainsi d'appliquer les reformulations sur les ressources du Web sémantique. Nous proposons par ailleurs, de développer un raisonneur à partir de cas fondé sur un ensemble de services Web réalisant les inférences nécessaires sur le modèle RDF(S) des reformulations. Une de nos perspectives prioritaires est la mise en œuvre effective de ces services pour appliquer le processus de RÀPC sur les ressources du Web sémantique utiles à l'application KASIMIR. Nous pourrions ainsi étendre le modèle des objets et le modèle des tâches proposés pour le RÀPC par l'implantation d'autres approches du RÀPC et d'autres fonctionnalités de représentation des connaissances. Il pourra s'agir par exemple de la représentation explicite de *points de vue* dont l'intérêt pour le RÀPC et l'application KASIMIR est décrit dans (d'Aquin *et al.*, 2002).

Remerciements

L'auteur tient à remercier les deux lecteurs anonymes pour leurs remarques judicieuses et leurs commentaires pertinents.

Références

BECHHOFFER S., HORROCKS I., GOBLE C. & STEVENS R. (2001). OilEd : a reason-able ontology editor for the semantic web. In *Proc. of the Joint German Austrian Conference on AI*, number 2174 in Lecture Notes In Artificial Intelligence, p. 396–408 : Springer-Verlag.

- BERNERS-LEE T., HENDLER J. & LASSILA O. (2001). The Semantic Web. *Scientific American*.
- BRACHAIS S., D'AQUIN M., LIEBER J. & NAPOLI A. (2003). Vers un Web sémantique en cancérologie. In *Journée Web sémantique médicale, WSM 2003*. Rennes. www.wsm2003.org.
- CHAMPIN P.-A. (2002). *Modéliser l'expérience pour en assister la réutilisation. De la Conception Assistée par Ordinateur au Web Sémantique*. Thèse d'université, Université Claude Bernard – Lyon 1.
- D'AQUIN M., LIEBER J. & NAPOLI A. (2002). Représentation multi-points de vue des connaissances pour l'adaptation. In M. C. JAULENT, Ed., *Actes du dixième atelier de raisonnement à partir de cas*, p. 23–31.
- DECKER S., VAN HARMELEN F., BROEKSTRA J., ERDMANN M., FENSEL D., HORROCKS I., KLEIN M. & MELNIK S. (2000). The semantic web : The roles of XML and RDF. *IEEE Internet Computing*, 4(5).
- FUCHS B. (1997). *Représentation des connaissances pour le raisonnement à partir de cas : le système Rocade*. Thèse d'université, Université Jean Monnet de Saint-Etienne.
- FUCHS B., LIEBER J., MILLE A. & NAPOLI A. (2001). Un algorithme pour la phase d'adaptation du raisonnement à partir de cas. In A. HERZIG, Ed., *Actes des journées nationales sur les modèles de raisonnement (JNMR'01)*, Arras, p. 79–92.
- HAARSLEV V. & MÖLLER R. (2001). Description of the RACER System and its Applications. In *Proc. of the International Workshop on Description Logics (DL-2001)*.
- HORROCKS I. (1999). FaCT and iFaCT. In *Proc. of the International Workshop on Description Logics (DL'99)*, p. 133–135.
- LASSILA O. (2002). Taking the RDF Model Theory Out for a Spin. In I. HORROCKS & J. HENDLER, Eds., *The Semantic Web - ISWC 2002, Lecture Notes in Computer Science 2342*, p. 307–317 : Springer Verlag.
- LAUBLET P., REYNAUD C. & CHARLET J. (2002). Sur quelques aspects du web sémantique. In *Actes des 2^e Assises nationales du GdR I³*, p. 59–78, Nancy.
- LIEBER J. (1997). *Raisonnement à partir de cas et classification hiérarchique. Application à la planification de synthèse en chimie organique*. Thèse d'université, Université Henri Poincaré Nancy 1.
- LIEBER J. (1999). Reformulations and Adaptation Decomposition. In J. LIEBER, E. MELIS, A. MILLE & A. NAPOLI, Eds., *Formalisation of Adaptation in Case-Based Reasoning : Third International Conference on Case-Based Reasoning Workshop, ICCBR-99*.
- LIEBER J., BEY P., BOISSON F., BRESSON B., FALZON P., LESUR A., NAPOLI A., RIOS M. & SAUVAGNAC C. (2001). Acquisition et modélisation de connaissances d'adaptation, une étude pour le traitement du cancer du sein. In J. CHARLET, Ed., *Actes des journées ingénierie des connaissances (IC-2001)*, p. 409–426, Grenoble.
- LIEBER J., D'AQUIN M., BEY P., BRESSON B., CROISSANT O., FALZON P., LESUR A., LÉVÊQUE J., MOLLO V., NAPOLI A., RIOS M. & SAUVAGNAC C. (2002). The Kasimir Project : Knowledge Management in Cancerology. In *Proc. of the 4th International Workshop on Enterprise Networking and Computing in Health Care Industry, HealthCom 2002*.
- MELIS E. (1995). A Model of Analogy-Driven Proof-Plan Construction. In *Proc. of the International Joint Conference on Artificial Intelligence, IJCAI 95*, p. 182–189.
- MELIS E., LIEBER J. & NAPOLI A. (1998). Reformulation in Case-Based Reasoning. In B. SMYTH & P. CUNNINGHAM, Eds., *Proc. of the 4th European Workshop on Case-Based Reasoning (EWCBR-98)*, Lecture Notes in Artificial Intelligence 1488, p. 172–183 : Springer.
- SINTEK M. & DECKER S. (2002). Triple—a query, inference, and transformation language for the semantic web. In *International Semantic Web Conference (ISWC)*, p. 364–379.
- SMYTH B. (1996). *Case-Based Design*. PhD. thesis, Trinity College, University of Dublin.